Generalized Formalization of Game Rules

Christiaan van de Sande Tanner Reese

September 16, 2020

1 Introduction

2 Rules

Definition 2.1. A rule on sets A and B, $R(A, B)$, is a triple named functions: $\lambda : A \to \mathcal{P}(B)$, $\mu : A \times B \to A$, and $\phi : A \times B \to A$, where $\mu(\phi(a, b), b) = a$.

The functions of a rule are responsible for replicating the mechanics of a game. The input set, A, is the set of all possible game states; the output set, B, is the set of all well formed moves that could apply to any game state. Then, λ maps a given game state to a set of legal moves on that game state; μ maps a game state and a move to a new game state, and ϕ.

3 Properties of rules

Definition 3.1. A rule R is repeatable iff $A_R \subseteq B_R$

Definition 3.2. Rules $R(A, B)$ and $S(A, C)$ are independent ($R \perp S$) iff, for all $a \in A$, $b \in B$, $c \in C$, $\mu_R(\mu_S(a, c), b) = \mu_S(\mu_R(a, b), c)$

4 Operations on Rules

4.1 Negation of a Repeatable Rule

$$S(A, A) = \neg R(A, A) \tag{1}$$

$$\lambda_S(a) = \begin{cases} \varnothing & \lambda_R(a) \neq \varnothing \\ \{a\} & \lambda_R(a) = \varnothing \end{cases} \tag{2}$$

$$\mu_S(a, b) = \mu_R(a, b) \tag{3}$$

$$\phi_S(a, b) = \phi_R(a, b) \tag{4}$$

4.2 Union between Rules

$$T(A, B) = R(A, B) \cup S(A, B) \tag{5}$$

$$\lambda_T(a) = \lambda_R(a) \cup \lambda_S(a) \tag{6}$$

$$\mu_T(a, b) = \mu_R(a, b) = \mu_S(a, b) \tag{7}$$

$$\phi_T(a, b) = \phi_R(a, b) = \phi_S(a, b) \tag{8}$$

4.3 Intersection between Rules

$$T(A,B) = R(A,B) \cap S(A,B) \tag{9}$$
$$\lambda_T(a) = \lambda_R(a) \cap \lambda_S(a) \tag{10}$$
$$\mu_T(a,b) = \mu_R(a,b) = \mu_S(a,b) \tag{11}$$
$$\phi_T(a,b) = \phi_R(a,b) = \phi_S(a,b) \tag{12}$$

4.4 Indetpendent Product between Rules

The independent product ($\times$) is only defined for rules that are Independent.

$$T(A,(B \times C)) = R(A,B) \times S(A,C) \tag{13}$$
$$\lambda_T(a) = \lambda_R(a) \times \lambda_S(a) \tag{14}$$
$$\mu_T(a,(b,c)) = \mu_R(\mu_S(a,c),b) \tag{15}$$
$$\phi_T(a,(b,c)) = \phi_S(\phi_R(a,b),c) \tag{16}$$

4.5 Dependent Product between Rules

The dependent product between two rules $S \cdot R$ is only defined if $A_S = B_R$.

$$T(A,(B \times C)) = R(A,B) \cdot S(A,B) \tag{17}$$
$$\lambda_T(a) = \{(b,c)|b \in \lambda_R(a), c \in \lambda_S(b)\} \tag{18}$$
$$\mu_T(a,(b,c)) = \mu_R(a,\mu_S(b,c)) \tag{19}$$
$$\phi_T(a,(b,c)) = \phi_R(a,\phi_S(b,c)) \tag{20}$$

4.6 Patterning of a Repeatable Rule over a Range

$$V(A,B) = R(B,B)\Big|_{S(A,B)}^{T(A,C)} \tag{21}$$
$$\nu_V(b) = \begin{cases} \{b\} & \lambda_T(b) \neq \varnothing \\ \{b\} \cup (\widehat{\nu}_V \circ \lambda_R(x)) & \lambda_T(b) = \varnothing \end{cases} \tag{22}$$
$$\lambda_V(a) = \widehat{\nu}_V \circ \lambda_S(a) \tag{23}$$
$$\mu_V(a,b) = \mu_S(a,\mu_R(b,b)) \tag{24}$$
$$\phi_V(a,b) = \phi_S(a,\phi_R(b,b)) \tag{25}$$

5 Implementation

6 Implications

6.1 Game Creation

6.2 Game Analysis

7 Conclusion